Metamorphosis

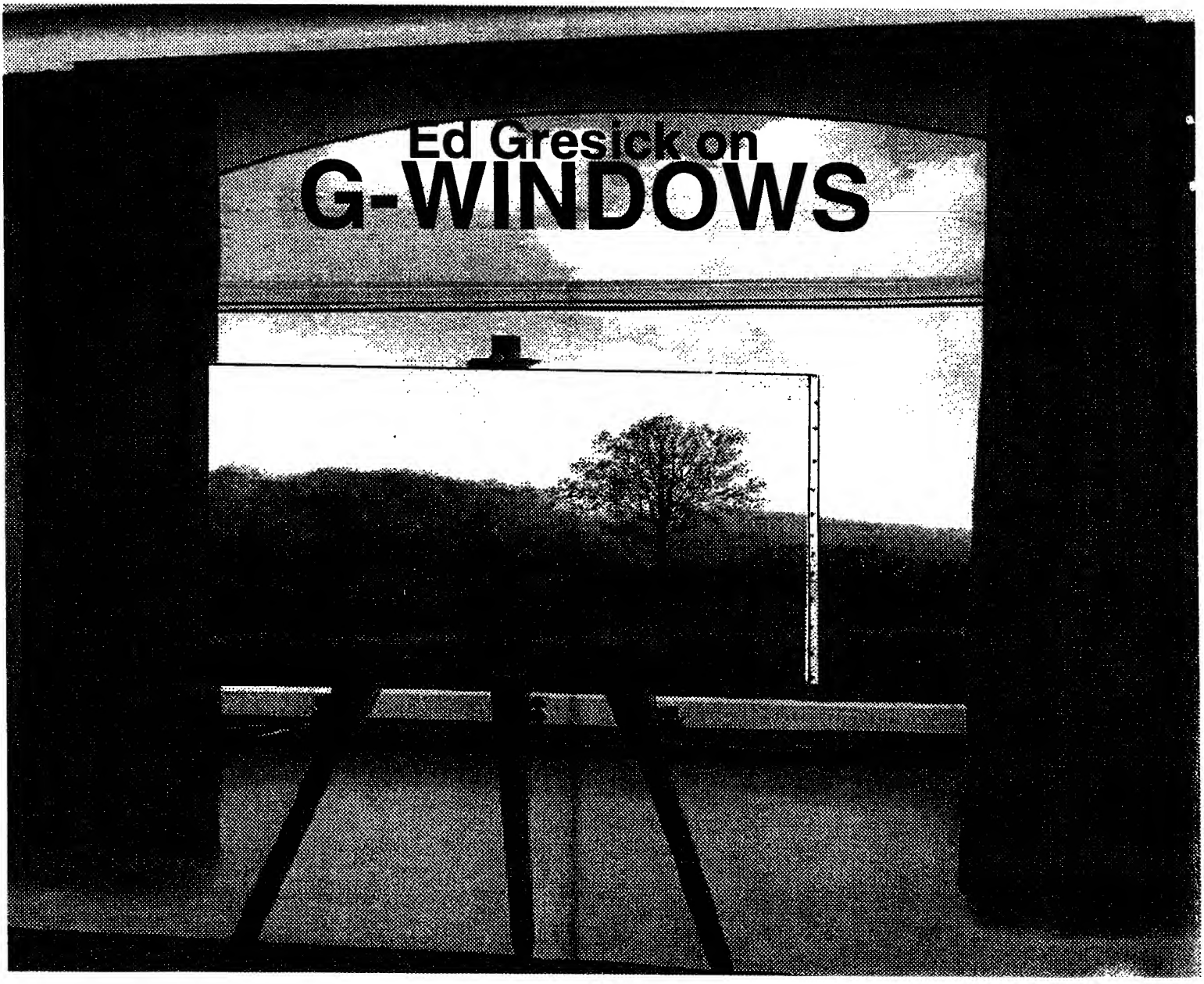
Into a New Era

Vol 1, Nbr 8 | For the CoCo/OS9/OSK Communities | February 1994

A Complete OS-9 File System Tutorial

Last of The Art of Programming Series

OS-9 on the Shuttle!



For superior OS-9 performance, the

SYSTEM V

Provides a 68020 running at 25 MHz, up to 128 MBytes of 0 wait-state memory, SCSI and IDE interfaces, 4 serial and 2 parallel ports, 5 16-bit and 2 8-bit ISA slots and much more. The SYSTEM V builds on the design concepts proven in the SYSTEM IV providing maximum flexibility and inexpensive expandability.

AN OS-9 FIRST - the MICROPROCESSOR is mounted on a daughter board which plugs onto the motherboard. This will permit inexpensive upgrades in the future when even greater performance is required.

G-WINDOWS benchmark performance index 0.15 seconds faster than a 68030 running at 30 MHz with ACRTC video board (85.90 seconds vs 86.05 seconds) using a standard VGA board.

Or, for less demanding requirements, the

SYSTEM IV

The perfect, low cost, high-quality and high performance OS-9 computer serving customers world-wide. Designed for and accepted by industry. Ideal low-cost work-station, development platform or just plain fun machine. Powerful, flexible and expandable inexpensively. Uses a 68000 microprocessor running at 16 MHz.

G-WINDOWS

NOW AVAILABLE FOR OS-9000

More vendors are offering G-WINDOWS with their hardware and more users are demanding it.

A PROVEN WINNER!

Available for the SYSTEM IV and SYSTEM V computers, the PT68K4 board from Peripheral Technology, the CD68X20 board from Computer Design Services and OS-9000.

OS-9/68000 SOFTWARE

CONTROLCALC - Real-Time Spread Sheet
SCULPTOR - Development and Run-Rime Systems
DATADEX - Free Form Data Management Program
VED ENHANCED - Text Editor
VPRINT - Print Formatter
QUICK ED - Screen Editor and Text Formatter
M6809 - OS-9 6809 Emulator/Interpreter
FLEXELINT - C Source Code Checker
IMP - Intelligent MAKE Program
DISASM_OS9 - OS-9 Disassembler
PROFILE - Program Profiler
WINDOWS - C Source Code Windowing Library
PAN UTILITIES

Distributor of Microware Systems Corporation Software

delmar co

PO Box 78 - 5238 Summit Bridge Road - Middletown, DE 19709
302-378-2555 FAX 302-378-2556

February 1994, Vol 1, Nbr 8

Contents

Metamorphosis
Published monthly by
The Dirt Cheap Computer Stuff Company
1368 Old Highway 50 East
Union, Missouri 63084

Copyright© 1993
All Rights Reserved

Publisher
Mark D. Griffith
Editor
Barbara Ann Griffith
Contributing Writers
Rodney Hamilton
Ed Gresick
Mark Griffith
Mike Guzzi
Shawn Marolf
Colin McKay

Subscriptions
Metamorphosis is published 12 times a year. Subscription rates are $24 per year for the continental United States. Canadian rates are $32 (US) per year. Overseas rates vary. Please write for details.

How To Contact Us
Mail can be sent to the address shown above. Electronic mail can be sent via the Internet to:

76070.41@compuserve.com
or
MARKGRIFFITH@delphi.com

Telephone calls can be made to (314) 583-1168. Please leave a message and your call will be returned.

This publication is composed, formatted, and master pages created entirely on machines running the OS-9 operating system.

Features

Columns

Departments

On the Front Cover

A "see-through" window display heralds our new column. This was a 676 x 854, 256 color GIF image converted to PostScript using **giftops.**

From The Editor's Desk

Here it is, the month of February, a month with many contrasts. This month started out with Ground Hog's Day. Those of us buried in the snow and wrapped up in a cocoon of coats, gloves, and mufflers poked our noses out to watch with anticipation to see just what our fate would be. Needless to say, we were not surprised by the outcome.

Later in the month Ash Wednesday was celebrated, and even Lincoln's and Washington's birthdays were mentioned. Some of us were lucky enough to get a paid vacation day from work for President's Day.

February in Missouri has been a month of below zero temperatures and driving in snow and ice. It has also been a month of good family evenings sitting wrapped up in a quilt in front of the fireplace with a big bowl of freshly popped buttered popcorn to watch rented movies. But February has been a lot more than that.

February means Valentine's Day, and we all know what that means: Cupid, red hearts, red roses, and, my favorite, chocolates! February is a month with dinner invitations, mushy cards, and couples saying I Love You. It is a month when many a young woman waits in nearly breathless anticipation to receive her engagement ring so she can spend the rest of the month giggling and showing it off to her friends. It is a month of reservations for a nice romantic dinner out with the one you love. February is a month full of love and warm fuzzies.

We were lucky enough to experience a couple of gorgeous spring-like days this February, as Mother Nature teased us with glimpses of what She has in store just a couple of months away. We had above normal temperatures on a couple of weekends, and it sure felt strange to be able to walk outside without being weighted down with protective gear. It also seemed strange to see people out in their yards washing the cars, playing with the dogs and kids, and having barbeques. Just seeing the sun was a treat this month.

Looking back, I think February was a good month. Sure it was cold and downright nasty outside, but inside glowed the embers of togetherness, fanned with love and gently stirred with caring. Can any month that inspires such feelings really be that bad?

Chicago Fest

As the time nears for the annual pilgrammage to Chicago for the *3rd Annual Last CoCoFest*, our thoughts are on ways we can represent ourselves to the Fest patrons. Too many other vendors are selling T-shirts and other items. We'd like to do something different, yet something that stands out amoungst the crowd. What would you suggest? Write in with your ideas and we'll see what we can do.

Mark will be there for sure, and I'll probably be able to make it this time. I'm looking forward to meeting everyone there. I'm especially looking forward to meeting Carl Kreider's wife, Debbie. From what Mark tells me, we'll be quick "Mall Buddies". Look out if you're at the local shopping mall!

Future Issues

While on the subject of the Chicago Fest, the 1st of May marks our one year anniversary. We'll have to do something special for the May issue this year We have ideas for some other future issues that might spice up your live. I won't disclose and details, but I hope you enjoy them. As always, we welcome any ideas our readers have for how we can serve you better. Please, if you have any suggestions, or would just like to send in a letter saying "thanks", please do so. Mark gets all the mail. I never get any for me (sniff).

New Family Addition

For those of you that know us, we have quite a number of pets. At last count, we had four dogs, three cats, three cockatiels, and a few mice that invaded the house (the cats are REAL interested in them). We recently had to put to sleep our oldest pet, a teacup poodle named "Scruffy". To keep from crying myself to sleep, Mark bought me a toy poodle puppy, and we named her "Poofy Bear", since that is what she looks like. We hope "Poofy" will be with us a long time.

Keep warm everyone! Spring is just around the corner!

Mail Call!

On Submissions

Dear Mark and Barbara:

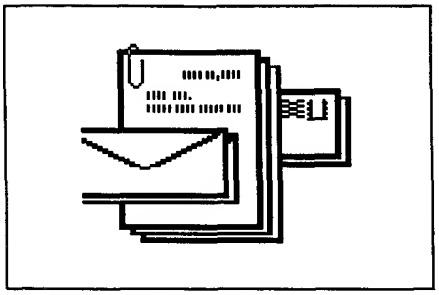
I am writing in response to your answer to the letter by Dave Gantz, titled *Editorial Rebuttal.* I have submitted two articles to your publication. The first you printed. Thank you again. The second was mailed by me on 26 December, 1993. I have yet to hear from you regarding the second one, but this does not bother me. I doubt that anyone is writing articles for the monetary gain regarding OS-9 on the CoCo, MM-1, System IV or what have you. A publication that would be interested in paying for these types of articles does not exist. I write for the love of it, to see my name in print, to help the OS-9 community, any number of things. I enjoy reading other's articles about OS-9 and hope someone enjoys reading mine. My only suggestion would be to let the author know that you have received the article as soon as possible. If a decision has already been made to publish or not to publish, let him know at that time. In any case, as soon as you decide to publish, even if an issue several months down the road, let the author know that you will publish. I would not submit my articles to another publication unless, and until, you tell me that you were not going to publish.

I would also like to tell you that the article by Mike Guzzi, OS-9 File Structure, is the best one I have ever seen explaining this subject. I myself was an electronic instructor for several years and have given several lectures about OS-9 on the CoCo at our club meetings. This was the best explanation I have ever read for the beginner in OS-9. I will refer to this article now any time I am trying to explain OS-9 File Structure to a fellow club member. An excellent article. Good show Mike.

Thanks again for publishing a very good magazine. I look forward to each and every one. Until next time...

David Wordell

We appreciate your comments David, and will try to do as you suggested. Just for the record, we will always print any article sent in to us, unless it is obscene, greatly inflammatory, or in otherwise poor taste. However, we will try to let every submitter know in advance when their article will be printed. Mike thanks you for your very kind comments. He does do a very good job, doesn't he?—Barbara



News Clips

More Machine Rumors

Not very much floating around concerning the new OSK machine being designed. However, I have learned that possibly three windowing systems will be available for this machine, G-WINDOWS, KWindows, and MGR. Which will be the default package has still not been decided.

BlackHawk Enterprises

David Graham, who recently bought the rights to continue producing the MM/1 computer, has been doing a bang-up job. He has orders for new machines and is working with various vendors to bring all the pieces together to be able to begin shipping to the new owners. In addition, David told me that a newly designed I/O board will be available soon that will correct many of the problems found in the earlier design. With this, David hopes to be able to continue selling MM/1's well into the future. Good luck David!

More on Article Submissions

Last month we posted a request for articles and a list of some ideas. So far, we have received only a couple responses. The life blood of this and any other magazine are articles. Without them, we would have nothing to print. As we move further along (this is the 8th issue), we are finding ourselves in dire need of something to print. Without submissions from the readers or any one else connected with the OS-9 or CoCo community, we will be forced to cease publication in the near future. Please, we need your support.

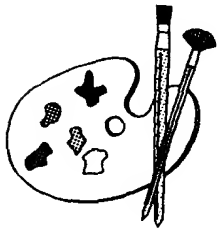
The Art of Programming

Part VII

The Phases of Programming

by Shaun Marolf

Shaun Marolf, 30, is a member of the US Navy stationed at Naval Air Station San Diego where he is currently working on his degree in Computer Sciences. He first learned programming in 1979 on an Apple II Plus and is also schooled in electronics and digital circuits. Shaun runs the "Eight Bit Heaven" BBS (619-447-2111) and owns several computers, including an original grey case CoCo 1.



This is the final installment in the series, and, as such, I want to say thank you to Mark and Barbara for running it. I also want to say thank you to all the people who helped me along the way, especially Harry Barner, my friend and electronics instructor. May God keep your soul. Thanks for putting up with all the silly questions.

When I first saw the name that Barbara had given to the series, I honestly thought it at first not to be right. However, after a few minutes of thought, I realized that it was extremely appropriate. I had never considered programming to be a form of art, yet it requires a great deal of creativity to conceive the concept of a program and to plan it out.

No two programmers work a program quite the same way, and even a simple task will be handled differently. Even if the end result of the program is the same, there is always some subtle difference between the two. It may not be noticed, but it is there none the less.

Programming is still a very technically orientated process and requires a high degree of logical thinking. We will go over the phases (steps) of programming.

1. Conception–We must first come up with an idea for a program and a loose idea of how we want it to work.
2. Output Design–Laying out the way we want the program to look to the end user will better help us draft the flowchart and write the code.
3. Input/Output Flowcharting–Though not a necessary step, many programmers find it to be a useful one. This phase allows the programmer to get a feel for how the program will work and will allow him to keep track of how the program interfaces with the user.
4. Optimization–Keep in mind the give and takes involved with writing a program. Decide where to enhance and where to sacrifice to best suit the program. The factors follow:
    a. Efficiency
    b. Execution Speed
    c. Programming Time
    d. Function
    e. Workability
    f. Reliability
    g. Recoverability
    h. Ease of Operation
    i. Ease of Training
    j. Capacity
    k. Portability
    l. Compatibility
    m. Maintainability
    n. Ease of Modification
    o. Documentation
    p. Attractiveness
5. Process Organizing–This step is for modular programming. Logically breakdown the separate processes and functions of the program to create the separate modules.
6. Operation Flowcharting–The flowchart of the actual program operations and data flow. Remember to follow the structure rules of the language in which you will be writing the program.
7. Coding–The actual program itself. Try your best to watch your syntax. This will help avoid errors.

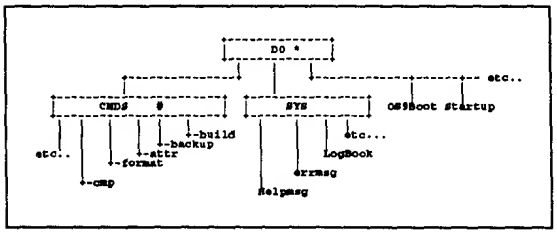
OS-9 File Structure

Part III

by Mike Guzzi

Mike Guzzi, 25, has a BS in Electrical Engineering, and is currently a technician at Specialty Records Corp/WEA Manufacturing INC. Mike wrote the APBBS BBS system that was sold by Second City Software. He has made several contributions to the public domain including Mbackup which allows a CoCo user to use more than 64k for copying a disk legitimately, under OS-9. Mike has owned a CoCo since 1984. He can be reached at **mike_guzzi@delphi.com.**

So far, we have gone over accessing files that contain data. Remember, I said that OS-9 stores most of its commands on disk. The principles of accessing commands (or executable files) are similar. Let's look at your system disk after you have booted os9.



You will see the familar asterisk "*" for the current working directory. Now let's concentrate on the pound sign, "#", that will indicate our current execution directory. This is where OS-9 will look to find files that contain programs that OS-9 can run when you type an OS-9 command, such as:

```
format /d1
```

OS-9 will first look for the module 'format' in memory. That's why we went over the module directory, or memory directory. If it does not find it in memory it will then try to load it off the disk using the current execution directory. If that fails it tries one more time by trying to find the file in the current working directory, but it will not consider it as a program, it will try to run it as a script file which I will explain below.

When you do a Mdir after booting OS-9, look at the bottom few lines. You will see names like copy, date, deiniz, del, dir, display, echo, and so on. These are OS-9 commands that are already in memory and will run without needing to load anything off the disk. So what do you do if you need to run a program that doesn't exist in memory or on the disk? For example you want to call up BASIC09 which is on the second disk you got with OS-9. Well, you could do one of two things. Remove your system disk, insert the Boot/config/basic09 disk, and type in the following commands:

```
chx /d0/cmds
chd /d0
```

Don't forget you have to change your working directory as well, since when you changed disks, both working and execution directory changed.

A second way to run basic09 is to put in the basic09 disk and type the following command:

```
load /d0/cmds/basic09
```

Now, replace the system disk in the drive. Since you used an absolute pathlist it followed that path to load basic09 without needing to refer to its current pointers. Basic09 will now be in memory and will run immediately when you call it up. Later, when you are finished, you remove it from memory by typing the following:

```
unlink basic09
```

I mentioned above about script files or shell scripts. They are files that contain text that OS-9 executes like if you had typed it in from the keyboard. A good example is a file called 'startup'. which is on the system disk. Put your system disk in drive 0 and type the following commands:

```
chd /d0
list startup
```

You will see the following:

```
* Echo welcome message
echo * Welcome to OS-9 LEVEL 2 *
echo * on the Color Computer 3 *
* Lock shell and std utils into memory
link shell
* Start system time from keyboard
setime </1
date t
```

This is called a shell script. Each line in this textfile is an OS-9 command line. You will notice in the first, fourth, and sixth line an asterisk as the first character, that tells OS-9 to ignore the line, it's used as a comment and is not executed. You can make your own shell scripts using the build command.

You will also notice in the seventh line some strange characters. The command 'setime' sets the clock to the current datetime, but what's this </1

part? That is called re-direction. When executing OS-9 commands there are three standard paths which it will use. They are called: Standard input, Standard output, and standard error.

Standard input: The place where OS-9 commands will get input for commands. Normally this will be the keyboard, if you were to type the following:

```
build sample.txt
```

Build will take whatever is typed on the keyboard and send it to the file called 'sample.txt' The same thing holds true for commands that ask for any keystrokes, like yes or no prompt for the format utility.

Standard Output: The place where OS-9 commands will display information. Normally this is the display screen. For example, the command:

```
list startup
```

This will list out the contents of the 'startup' file to your screen.

Standard error: The place where OS-9 commands will display error messages if something goes wrong. Nomally this is also the display screen.

The shell allows us to re-direct or change where input or output goes to, by using the symbols:

```
<    to re-direct input
>    to re-direct output
>>   to re-direct error output
```

Say you wanted to list your startup file to your printer. Normally the list command will display it on your screen, but if we do this:

```
list startup >/p
```

This will list the startup file, but instead of displaying it on the screen the >/p part tells OS-9 we want to take the output and send it to a device called /p which is the printer. If there was an error generated with the list command, you would see it on your screen, that's why there is a seperate path for error messages. You can also combine these operators as such:

```
<>      re-directs input and output
<>>     re-directs input and error
>>>     re-directs output and error
<>>>    re-directs input, output, and
error
```

The reason you see 'setime </1' in the startup file is that normally setime would get its input from the keyboard, but since the program was called from a script file, the input is the file. The </1 tells OS-9 that we want setime to get its information from the keyboard. OS-9 can get rather complex with having these options. Let's look at an imaginary situation.

Say you have a program that will will look up names and will display their phone number. The program is called 'phonenum'. It gets the names from the keyboard and will display the namephone number of that person. You can create a textfile with these names in it and use that for input for this program. In addition, you want a file with these namesnumbers for later reference. After creating the list of names, let's call it 'names', we will send it to a file called 'numbers'. Here is what the command line would look like.

```
phonenum <names >numbers >>notfound
```

Phonenum will execute taking the names from the file 'names', look them up, and send the names and numbers to a file called 'numbers'. Any names it cannot find would normally be displayed on the error path, but we sent it to a file called 'notfound'.

Another way to manipulate input and output is by using pipes. What this means is to link two or more programs so the output of one will become the input for the next one. Let's use the example above, but use a pipe.

```
list names ! phonenum >numbers >>not-
found
```

Pipes come in handy for programs that take data from the standard input, modify it in some way, and then send the data to standard output. Well, that's enough of pipes and standard paths. Let's look at some things you may encounter while using OS-9 and how to identify files.

The ERROR and HELP utility

When you do get an error message, you will want to know what the number stands for. You can look it up in the OS-9 book, but there's a better way. With your system disk in the drive. you can find out what error number mean like this:

```
error 216
```

You will see the following output.

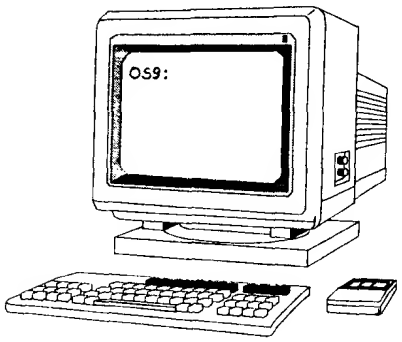
```
216 - Path name not found
```

This tells you that error code 216 means OS-9 could not find the file you requested. You may include more then 1 error code on the line like this:

Gee, Windows!

by Ed Gresick

Ed Gresick, (age unknown), owner of **delmar co.,** *has been an avid user and advocate for OS-9 since its beginnings. He is sometimes known as OS-9's only Historical Monument. Ed can be reached through this magazine and at* **edelmar@delphi.com** *or* **76576.3312@compuserve.com**



This is the first of a series of articles about G-WINDOWS. This installment will cover the history, objectives, organization and a few features of G-WINDOWS. Future articles will describe how to use G-WINDOWS. If there is sufficient interest, this series can continue with programming under G-WINDOWS with the Developer's Pack.

Much of the information to be presented will come from the section titled **GETTING STARTED** from the *G-WINDOWS User's Manual* distributed with G-WINDOWS by **delmar co.**

HISTORY

About 6 years ago, Steve Adams felt OS-9 needed a windowing system. At the time, he was working with graphics for a Company providing VME equipment. He investigated X-Windows being developed at MIT. Reviewing this code, he decided it would be better to follow his own knowledge of graphics and OS-9 and his instincts.

Steve purchased an Atari 1040 and went to work. Many of his concepts are original, but many of the features have roots in the work done by the Xerox Corporation at their Pala Alto facility (PARC) in the early 80's as have most of the popular windowing systems including X-Windows. Steve's work came to the attention of GESPAC, an International Company based in Switzerland, one of the leading providers of OS-9 hardware. GESPAC was looking for a windowing system and a marriage was made. G-WINDOWS has been available commercially since 1990.

OBJECTIVES

Some of the objectives in designing G-WINDOWS were

- multiple, active, text/graphics windows to take full advantage of OS-9's multi-tasking capabilities
- support for pointing devices such as mice and touch screens
- ease the use of the more common OS-9 commands
- insure applications written under one platform will work on all platforms supporting G-WINDOWS
- support for screen resolutions from 640 x 480 pixels and higher
- support 2 colors (monochrome), 16 colors and 256 colors
- small code size to minimize resource requirements
- permit easy porting to almost any OS-9 platform

ORGANIZATION

As distributed, G-WINDOWS comprises 2 main components: WFM, the window file manager and DESKTOP, the Graphical User Interface.

WFM is central to G-WINDOWS and permits the creation and control of multiple, varying sized and types of windows on a graphics screen. Keyboard and mouse (or other pointing device) inputs are routed to the selected process by WFM. WFM supports pop-up menus, alert messages and request boxes when required by an application. A process may have more than one window and access them at will. WFM expands the command line editing capabilities of OS-9, and it will store the last 50 commands which may be recalled with the arrow keys. Two classes of fonts are supported by WFM. Quick fonts (qfonts) provides three user selectable font

sizes. The second class of fonts, the general fonts, provide several different font types, Courier, Helvetica and Times in sizes ranging from 9 pts to 72 pts.

The DESKTOP Manager is an application which runs under WFM. DESKTOP replaces many of the file handling and program starting functions of the OS-9 shell. DESKTOP is not intended to replace the shell as an execution or development tool, but may be used in conjunction with shell. It is visually oriented rather than text oriented. Most DESKTOP commands are entered by pointing and clicking a pointing device - usually a mouse. Several other features are included to ease usage. A custom, user configurable menu is supported. The custom menu permits selection of processes by pointing and clicking the mouse. Additionally, many file types are recognized permitting the user to point to the file name or icon and click the mouse to start the appropriate process.

It is not necessary to run DESKTOP when using G-WINDOWS. Indeed, there are many instances, mostly in the industrial/commercial markets, where an application will be used and the user will never leave it.

UTILITIES & DEMOS

There are many utilities and demo programs provided with most G-WINDOWS distribution diskettes. Some of these are -

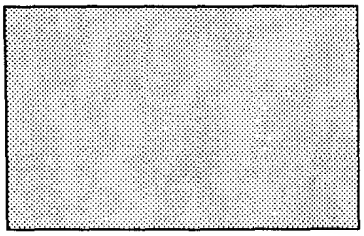
calc	a simple calculator - does decimal to hex to binary conversions
clock	default is small and unobtrusive
compress_image	saves disk storage space
cpu_usage	shows actual cpu usage graphically
edit_8bit	a simple graphics editor
edit_64	similar to edit_8bit but restricted to a small image
export_gif	export a G-WINDOWS image file to a .gif file
eyes	tells you where the mouse is
fns	fish and sharks demo program
font_demo	displays the different font styles and point sizes available
image_to_deskjet	print an image to a deskjet printer
image_to_paintjet	print an image to a paintjet printer
import_gif	import a .gif file to G-WINDOWS image format
maze	nice demo
mem_usage	shows memory usage - especially useful when memory is tight
menu_demo	demonstration of menuing capabilities of G-WINDOWS
savecrt	save all or selected portions of a G-WINDOWS screen/window
viewimage	view a G-WINDOWS image file

Next month we'll get into using G-WINDOWS and the benefits it may offer the user.

There are several windowing systems available for the OS-9 operating system. The first one commercially available (late 1986) was Multi-Vue for OS-9 Level II running on the Radio Shack Color Computer 3. G-WINDOWS came out shortly thereafter. Another package that is freely distributed via the computer networks is MGR. Kwindows is a system available only on the MM/1 computer and is a work-a-like of the Multi-Vue system.

Windowing systems have not been very popular with OS-9 users for a number of reasons. The most common complaints have been lack of speed and being limited to what the window manager wants to let you do. Many window system users do nothing more than open one or more command line windows and use each as a virtual machine. Some users cannot manipulate a mouse accurately enough. Others resort to using keyboard macros and avoid the mouse altogether. Getting users to work with the system can be very difficult indeed.

As the rest of the computing world moves toward windowing systems as the preferred environment, the challenge is to make systems like G-WINDOWS flexible and fast enough to please the users. The benefits are hard to describe, but as many one-time windows nay-sayers will tell you, once you make the conversion, you'll never go back.



Straight From The Horse's Mouth

by Colin McKay

Colin McKay is the Executive Vice-President of the OS-9 Users Group, Inc. He writes a regular column for the Users Group Newsletter, the MOTD. The column, "Straight From the Horse's Mouth", is about the use of OS-9 in Educational, Industrial and Scientific institutions. The following article is a reprint from that column. This article was written by David Stang, a software engineer working for the Space Experiments Department of NASA's Lewis Research Center, and covers an experiment scheduled to fly on the Space Shuttle in March, 1994.

This article appears in place of this month's 'Hardware, from the Inside Out' column, which will be back next month.

Space Experiments Department
NASA Lewis Research Center

SOLAR ARRAY PLASMA INTERACTION EXPERIMENT (SAMPIE)

SAMPIE is a space experiment being developed at the NASA Lewis Research Center in Cleveland, Ohio. It is scheduled to fly on the STS-62 Space Shuttle mission in March 1994.

The objective of SAMPIE is to investigate the electrical behavior of space materials when they are exposed to the space environment, a plasma consisting of high energy charged particles.

There are three kinds of adverse interactions between space materials and the plasma: current collection, snapover, and arcing. Current collection occurs on a solar array when high energy electrons or ions are picked up from the plasma. Snapover occurs at a highly positively biased conductor, and causes a great amount of current collection from the plasma. These currents appear as losses in array operating current and reduce the capability to produce power. Arcing occurs on highly negatively biased solar arrays and other surfaces containing conductor-insulator junctions. This leads to disruptions in power and electromagnetic interference.

SAMPIE's measurements will provide data for the verification of computer models, and help in the design of future spacecraft power systems.

The experiment hardware consists of an experiment plate, the electronics unit, plasma-potential and shuttle-bias probes, and a plasma pressure gauge. The experiment plate consists of different solar array samples and of small samples of various conducting materials and surface treatments. SAMPIE will be attached to a carrier riding in the orbiter's payload bay.

The electronic components are: two high voltage power supplies to bias the solar cells, a custom designed electrometer for highly accurate current collection, a custom designed transient current detector (TCD) for arc measurement, relay boards, I/O, plasma-measurement electronics, A/D boards, a single board computer (Motorola 68030 processor, RAM and ROM), and flash memory boards for data storage.

The VME bus was used. Since the system is required to run in a vacuum, the boards were built with thermal planes for conductive heat transfer. The computer, A/Ds, and Memory were purchased from DY-4 Systems, who specialize in Military and ruggedized VME products.

During the experiment a range of bias voltages is applied to each sample successively while current collection, arc characteristic, and plasma data are collected. After each sample is done the data is saved to the flash memory. The data will be reduced and analyzed after it returns to the ground.

The tasks of the software are to:

1. Control the high voltage power supplies and the relays.
2. Acquire data from the electrometer, TCD, and A/D boards.
3. Store the data onboard.
4. Communicate with the ground.

The software was written using Professional OS-9 V2.4 and Microware's PC-hosted development system PCBridge (C compiler, assembler, linker, and serial communication). During development, the executables were downloaded to the target using Kermit. The final version was programmed onto the ROMs.

OS-9 was used for its multitasking, interprocess communication, and timing capabilities. Data memory modules were used to share data between

processes. Drivers were written to handle some of the custom board data acquisition and to store to the Flash Memory.

About the Author

David Stang works for the Aerospace Design and Fabrication Corp., supporting the NASA Lewis Research Center in Cleveland. He is currently working as a software engineer for the Space Experiments department. His past projects for NASA have included work on a chemical vapor reactor which produces research semiconductor material, and instrumentation and image processing tools for non-destructive testing. He has a BS in Physics from Miami University, Oxford Ohio, and an MS from Penn State University.

David can be reached at:

David B Stang
2001 Aerospace Pkwy
Cleveland, OH 44142

If you are interested in joining the OS-9 Users Group and receiving the MOTD, please write to the address below.

The OS-9 Users Group, Inc.
6158 W 63rd St, Suite 109
Chicago, IL 60638, USA

Membership fees for one year are:

US or Canada - $25.00 US
All other countries - $30.00 US

Support OS-9!

AniMajik Productions

Hardware:

Keyboards, AT/XT switchable for your Sys4 or Sys5, MM/1 or Kix or even PC Compatible. Extremely small footprint. $59.95

C Micro Charts, C functions and syntax at your fingertips, spill-proof laminated plastic $6.50

Software:

For CoCo 512K, OS-9 Level 2 Machines

DCom (Basic09 DeCompiler) *Now shipping* V3.1 ..... $24.95

TShell (5 times faster than Multi-Vue) Text-based "GUI". Displays more info on your screen than MV does $19.95

Cloud_09 (Graphics/Animation Pakg) Save/Edit/Load VEF Pictures and Animate them. $19.95

For OSK Machines:

OSK ToolKit (Utility Pak for OSK) . $9.95

Underground Subscribers take 20% Off these low prices!* Include your member card number when you order.
*(Except Keyboards)

(Check or M.O., sorry no plastic or COD)

Please include $1.50 S&H with your order

AniMajik Productions
4650 Cahuenga Blvd, Ste #7
Toluca Lake, CA 91602
(818) 761-4135 (voice)
(818) 365-0477 (fax)

On The Lighter Side

How Many Pro Football Teams Can You Identify?

Now that the football season is over, can you remember all the teams? See how good your memory is with this quiz.

1. 747's _____
2. Stupid Sunbather _____
3. Wise Sunbather _____
4. $1.00 for Corn _____
5. Midnight Snackers _____
6. 6 Rulers _____
7. Uncles Spouse in the Army _____
8. Peter & Paul _____
9. Ewe's Mate _____
10. Half Bovine-Half Man _____
11. Marine Bird _____
12. Loaders _____
13. Seven Squared _____
14. Indians Leader _____
15. Credit Card Users _____
16. Henry's First Compact _____
17. A Girl's Toy and a Fish's Arm _____
18. Lubricators _____
19. American Forefathers _____
20. Fundamental Laws _____
21. I. O. U.'s _____
22. Six Shooters _____
23. Thieves _____
24. Streakers _____
25. King of Beasts _____
26. Tigers _____
27. U. S. Birds _____
28. Equine Goes Wild _____

Send in your answers and we'll post the names of those that get them all correct!

Dirt Cheap Computer Stuff Company

"Cheap, but not trash"

Modem Madness Sale!!

US Robotics Sportster 14.4FAX - $189.95

US Robotics Sportster 14.4 (no FAX) - $179.95

LineLink 14.4FAX - $139.95

Each of these modems support V.32/V.42bis, MNP Level 5, and send/receive FAX capability (except where noted). With transfer speeds up to 57,600 BPS, your online time will really fly!

Please call 314-583-1168 for more details. Also call about our other software and hardware products for OS-9 and OSK, such as ~~*InfoXpress*™~~, *Speedisk,* **CIRCAD,** Floptical drives, CDROMs, and more. All orders add $5.00 for shipping. Check or money order only please.

Dirt Cheap Computer Stuff Co.
1368 Old Highway 50 East
Union, Missouri 63084

Address Correction Requested